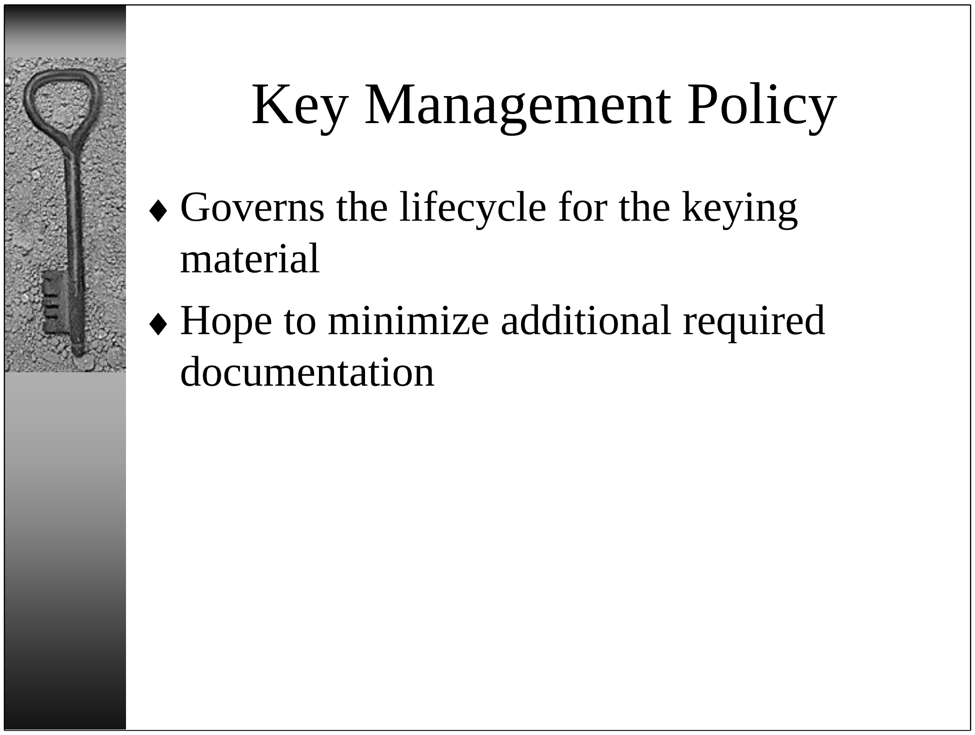
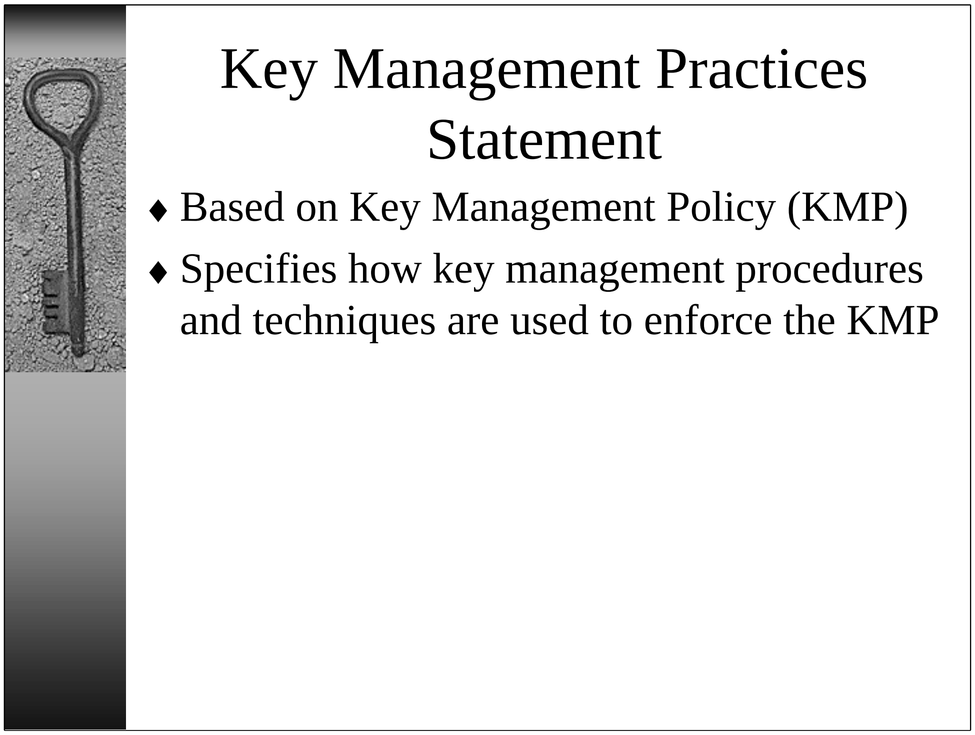
General Key Management Guidance



Key Management Policy

- ♦ Governs the lifecycle for the keying material
- ♦ Hope to minimize additional required documentation



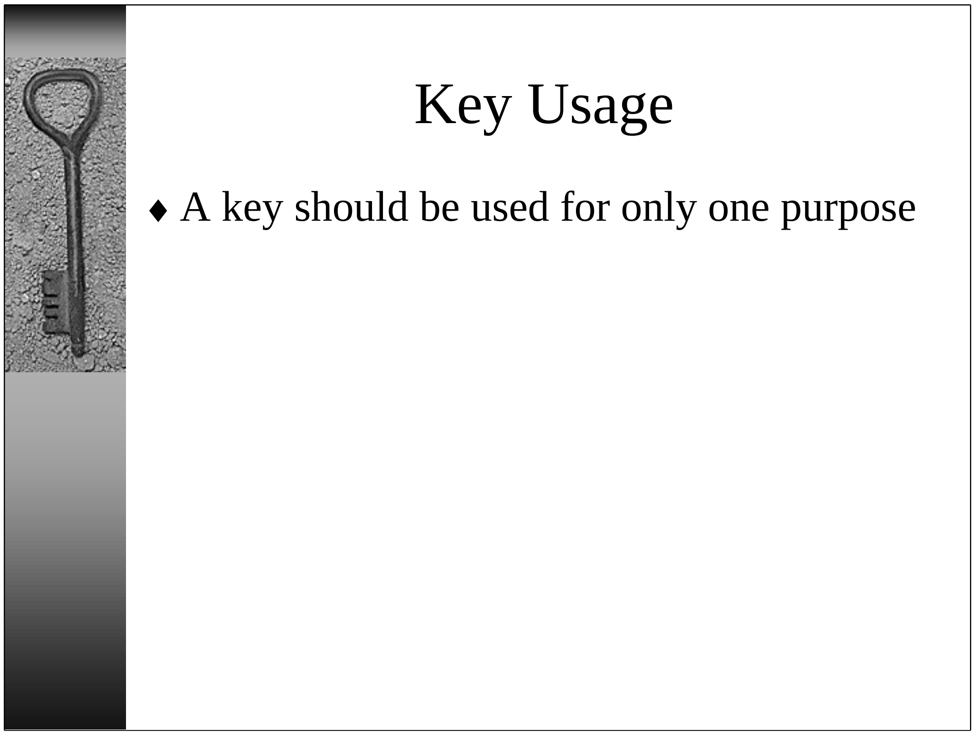
Key Management Practices Statement

- ♦ Based on Key Management Policy (KMP)
- ♦ Specifies how key management procedures and techniques are used to enforce the KMP

As part of the key management policy, we see the need for the development of a Key Mgmt. Practices Statement. The Practices Statement would be based on the Key Mgmt. Policy and would specify how key mgmt. procedures and techniques would be used to enforce the policy.

Example:

Key Mgmt. Policy statement might be that secret and private keys must be protected from unauthorized disclosure. Key Mgmt. Practices statement might say that secret and private keys must be either encrypted or physically protected.



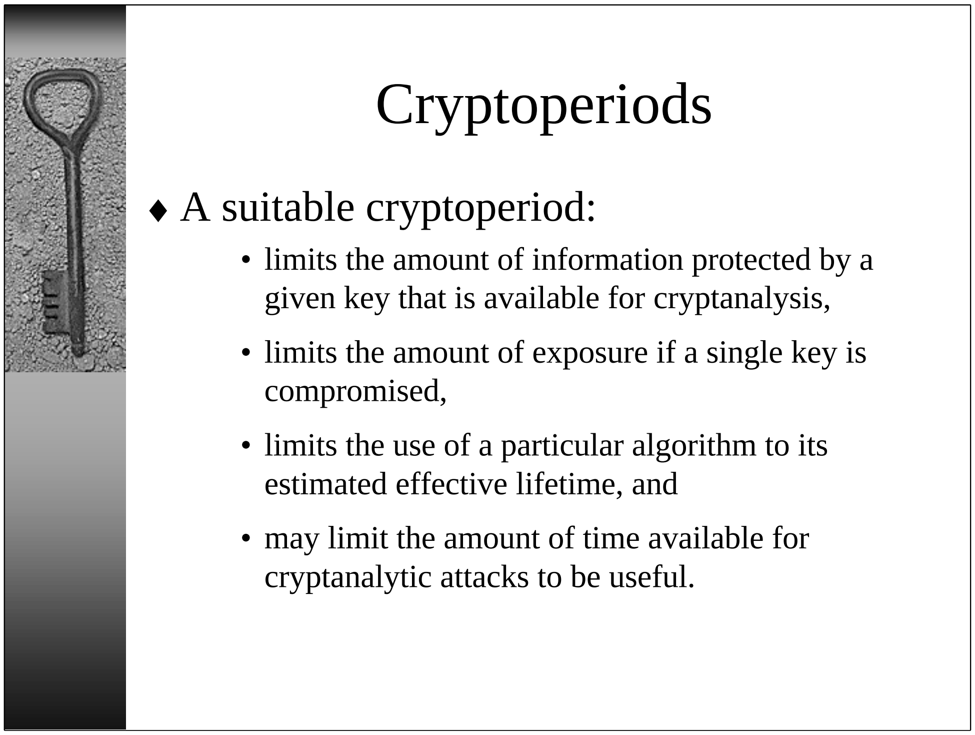
Key Usage

- ♦ A key should be used for only one purpose

In the key usage section, we’ve required that a key should be used for only one purpose.

For example:A symmetric key could be used for encrypting keys, OR encrypting data, OR generating a MAC, but no more than one of these.

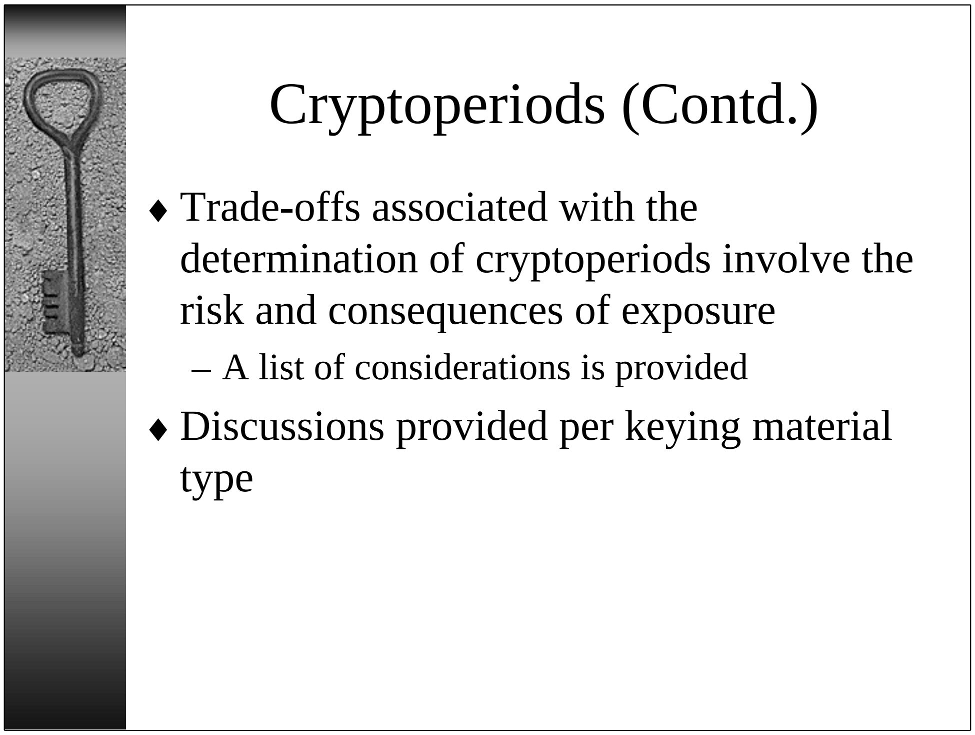
A key pair could be used for signing and verifying keys, Or for key establishment, but not both.



Cryptoperiods

- ♦ A suitable cryptoperiod:
  - limits the amount of information protected by a given key that is available for cryptanalysis,
  - limits the amount of exposure if a single key is compromised,
  - limits the use of a particular algorithm to its estimated effective lifetime, and
  - may limit the amount of time available for cryptanalytic attacks to be useful.

A cryptoperiod is the timespan during which a specific key is authorized for use. A suitable cryptoperiod...



Cryptoperiods (Contd.)

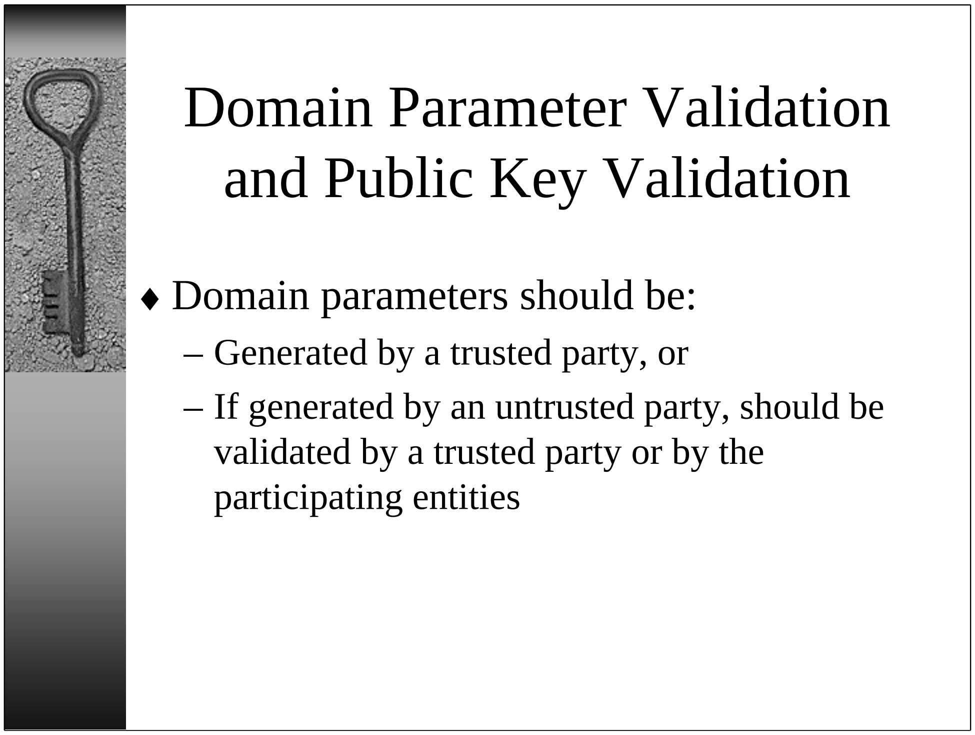
- Trade-offs associated with the determination of cryptoperiods involve the risk and consequences of exposure
  - A list of considerations is provided
- Discussions provided per keying material type

Tradeoffs…

A list of considerations is provided. For example:

- the strength of the cryptographic mechanism
- the volume of information flow or the number of transactions
- the security function (e.g., encryption, digital signature)
- the threat to the information

Initial discussions for the various types of keys have been provided.



Domain Parameter Validation and Public Key Validation

- ♦ Domain parameters should be:
  - – Generated by a trusted party, or
  - – If generated by an untrusted party, should be validated by a trusted party or by the participating entities

We have included a section on domain parameter and public key validation.The DSA, DH, and MQV algorithms are defined with domain parameters.

DSA and D-H: (p, q, g)

ECDSA and EC key establishment:

-- field size (q)

-- basis (FR)

-- the equation for the curve (a,b)

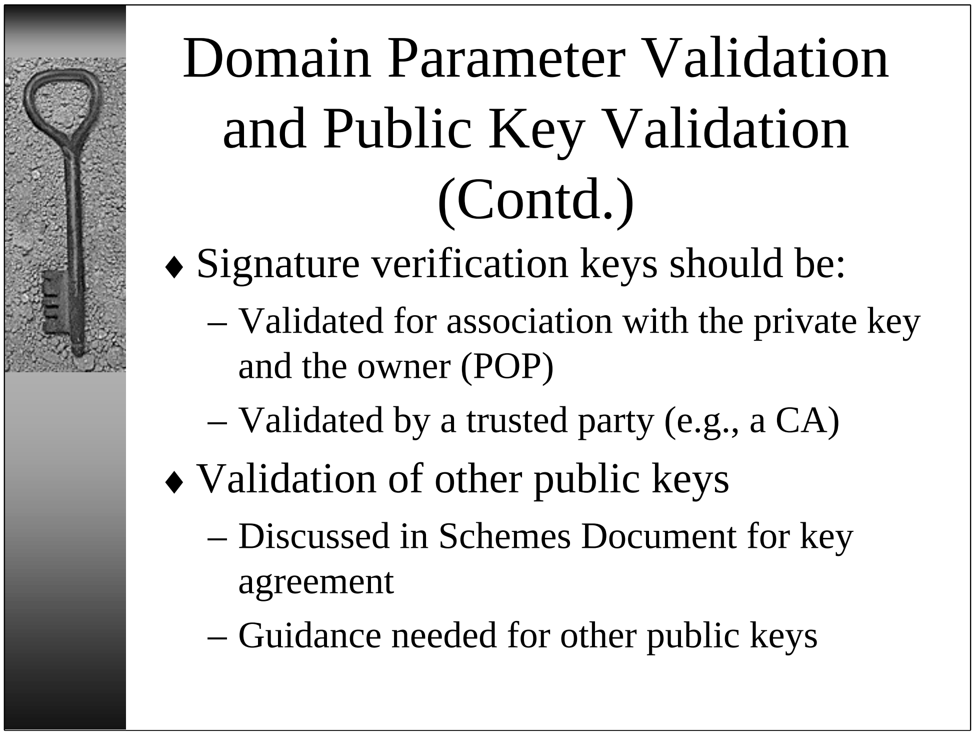
-- optional *SEED*

-- a point G on the curve

-- the order of the point G (n)

-- the cofactor h (the order of the curve/n)

Domain parameters should be ...



Domain Parameter Validation and Public Key Validation (Contd.)

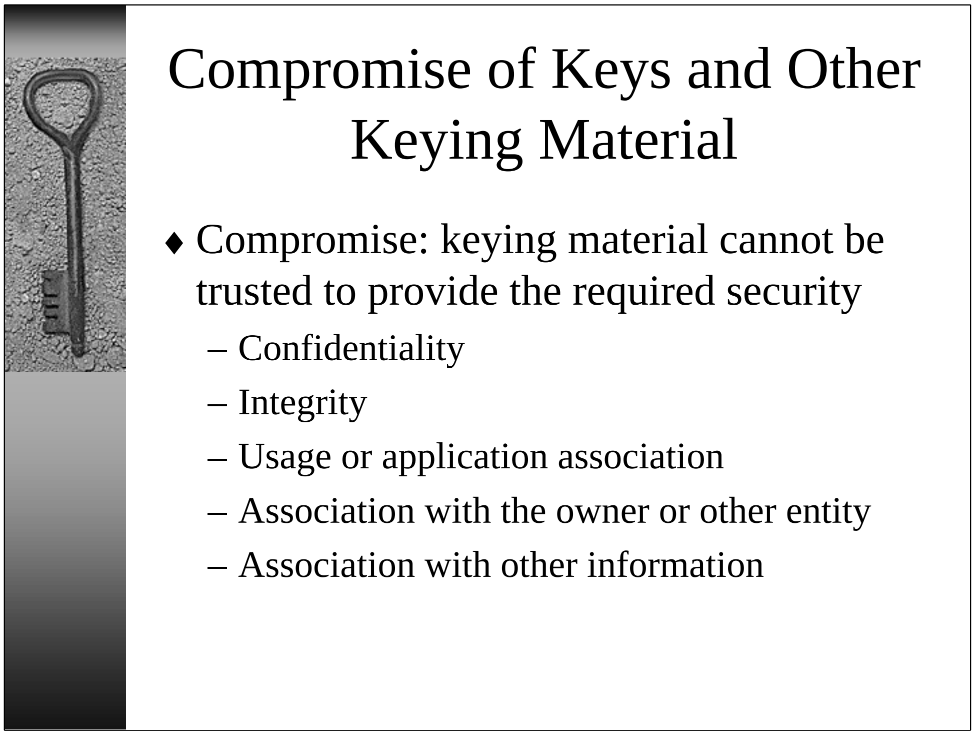
- ♦ Signature verification keys should be:
  - – Validated for association with the private key and the owner (POP)
  - – Validated by a trusted party (e.g., a CA)
- ♦ Validation of other public keys
  - – Discussed in Schemes Document for key agreement
  - – Guidance needed for other public keys

Signature verification keys…

The validation of other public keys…

The validation for signature verification keys and static key establishment public keys could be “attested to” in a certificate.

The validation of ephemeral public keys must be performed by the receiving entity.

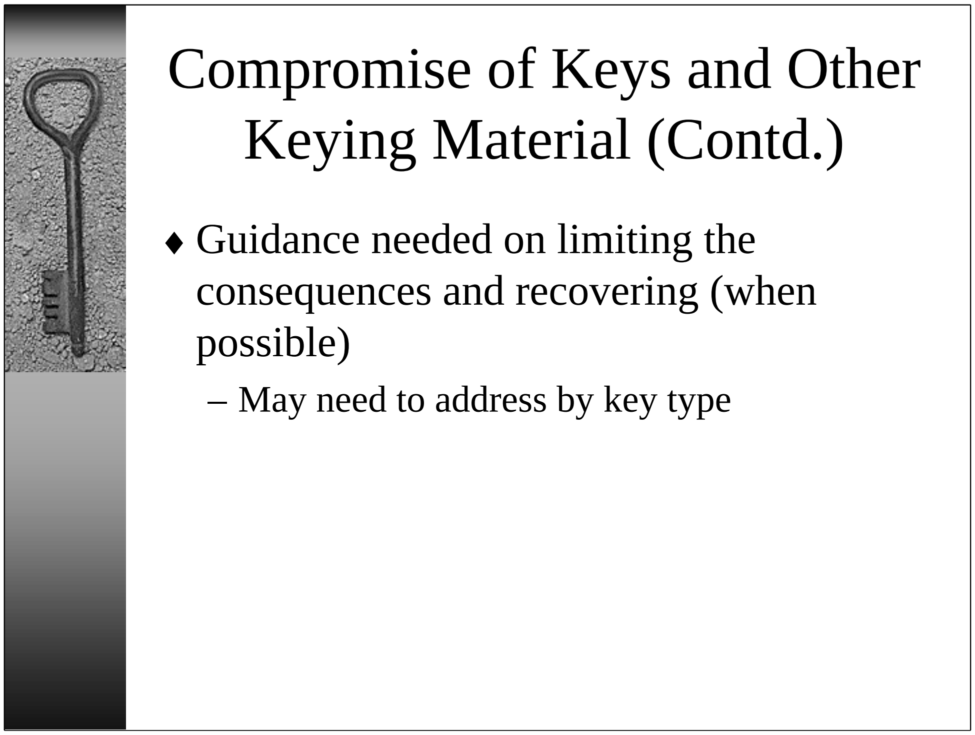


Compromise of Keys and Other Keying Material

- Compromise: keying material cannot be trusted to provide the required security
  - Confidentiality
  - Integrity
  - Usage or application association
  - Association with the owner or other entity
  - Association with other information

Cryptographically protected info is secure only if
the algorithm is strong, and the keys are not
compromised. Compromised keying material…
Keying material can be compromised in a number of ways:
If the ____ is compromised,

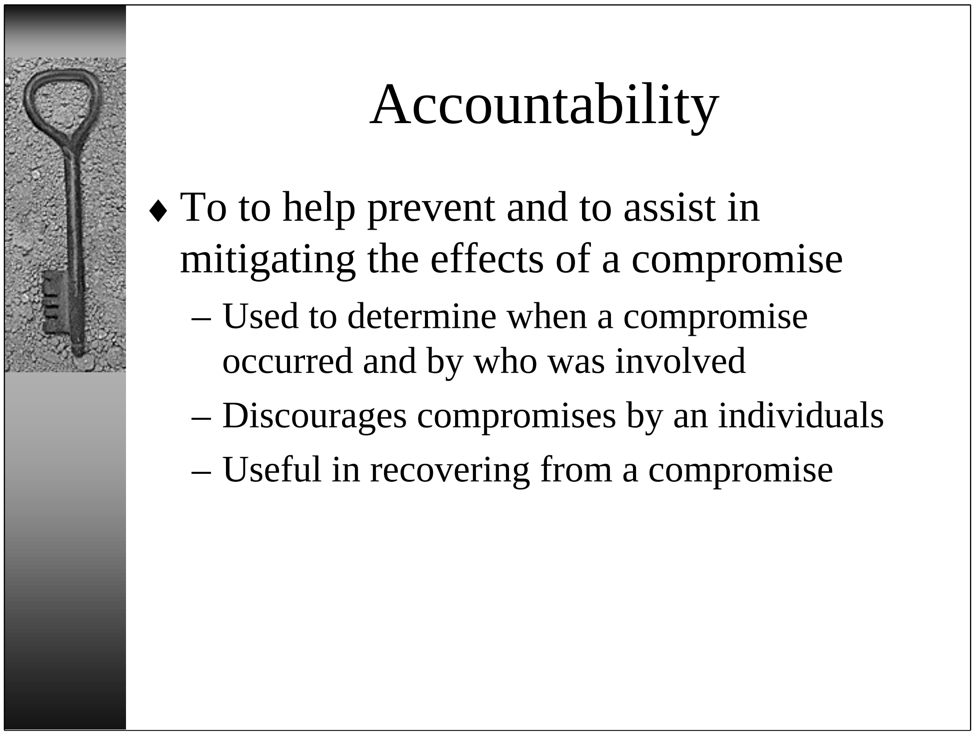
- another entity can use the key
- the key is incorrect
- the key could be used for the wrong purpose
- the identity of the other entity is unsure, or the info. cannot be processed correctly
- you can’t perform the cryptographic process correctly



Compromise of Keys and Other Keying Material (Contd.)

- Guidance needed on limiting the consequences and recovering (when possible)
  - May need to address by key type

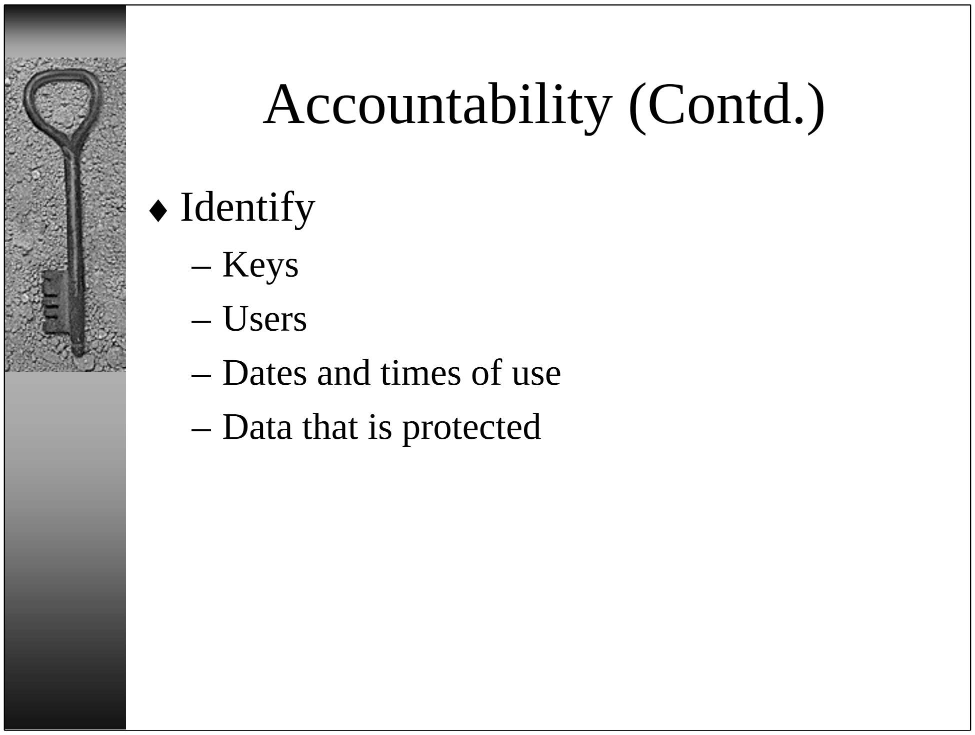
We need to also provide guidance on recovering from a compromise. This may need to be addressed by key type and type of compromise.



Accountability

- ♦ To to help prevent and to assist in mitigating the effects of a compromise
  - – Used to determine when a compromise occurred and by who was involved
  - – Discourages compromises by an individuals
  - – Useful in recovering from a compromise

During a key’s lifetime, the may be available somewhere in PT; it may reside in several systems; it may be under the control of several individuals; and it may be used to protect other keys.
A compromise is bound to happen sometime. Accountability can be an effective tool in preventing compromises and reducing the impact of detected compromises.



Accountability (Contd.)

- Identify
  - Keys
  - Users
  - Dates and times of use
  - Data that is protected

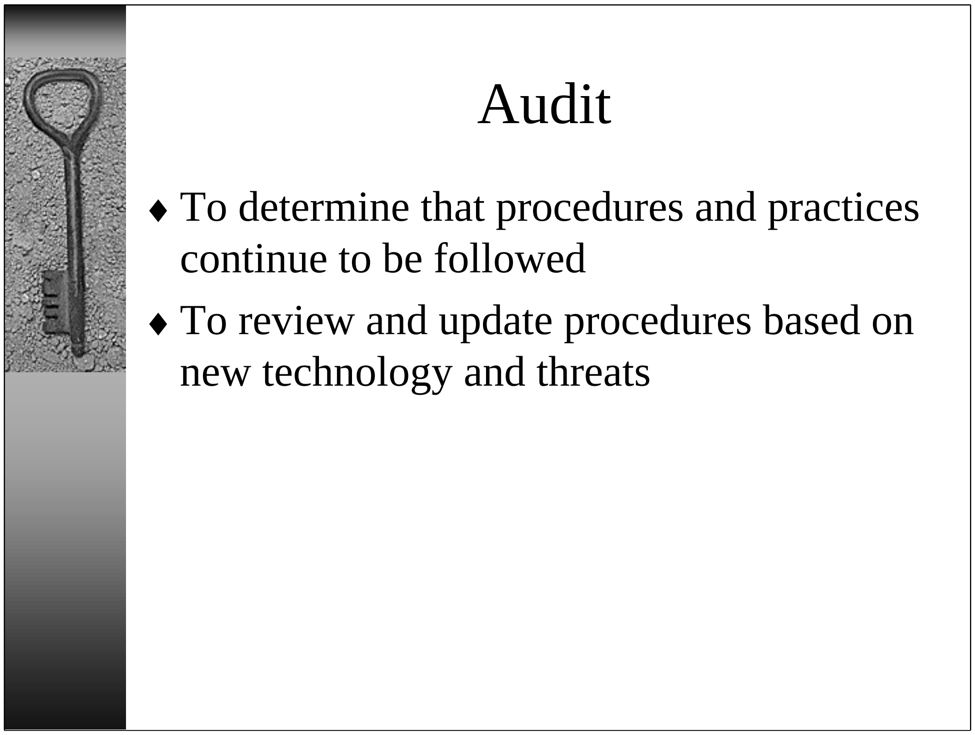
Accountability requires

-- the unique identification of keys

-- identifying who has access to the keys

-- identifying the dates and times of key usage

-- associating a key with the data it protects

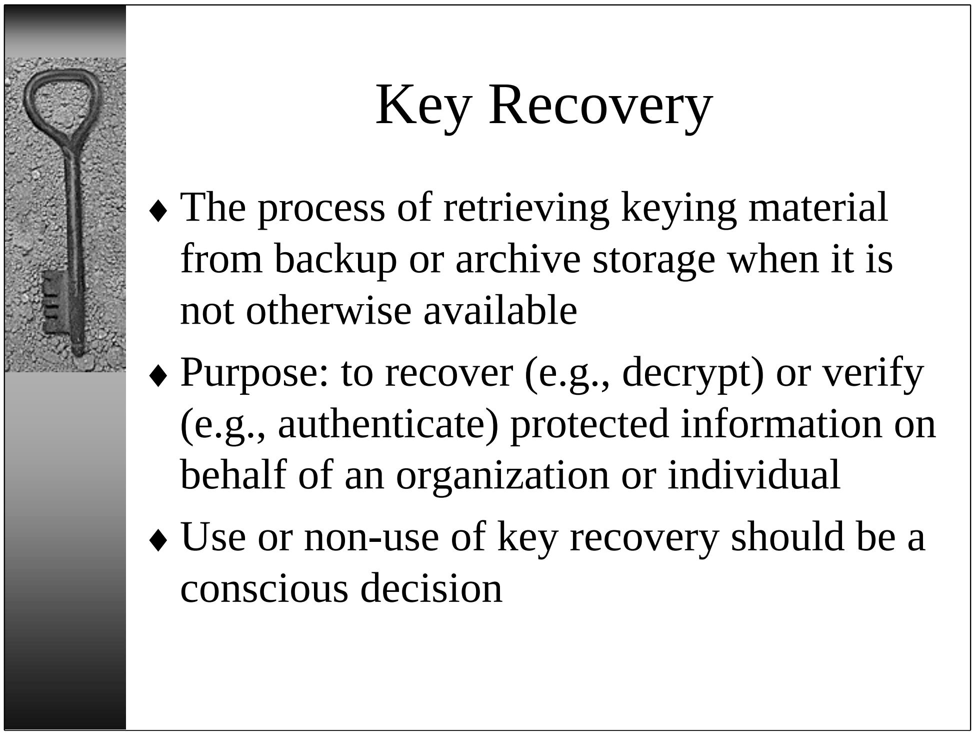


Audit

- To determine that procedures and practices continue to be followed
- To review and update procedures based on new technology and threats

Cryptographic systems can be compromised by lax or inappropriate practices or procedures.

KMS should be audited to determine that the practices continue to support the policy.



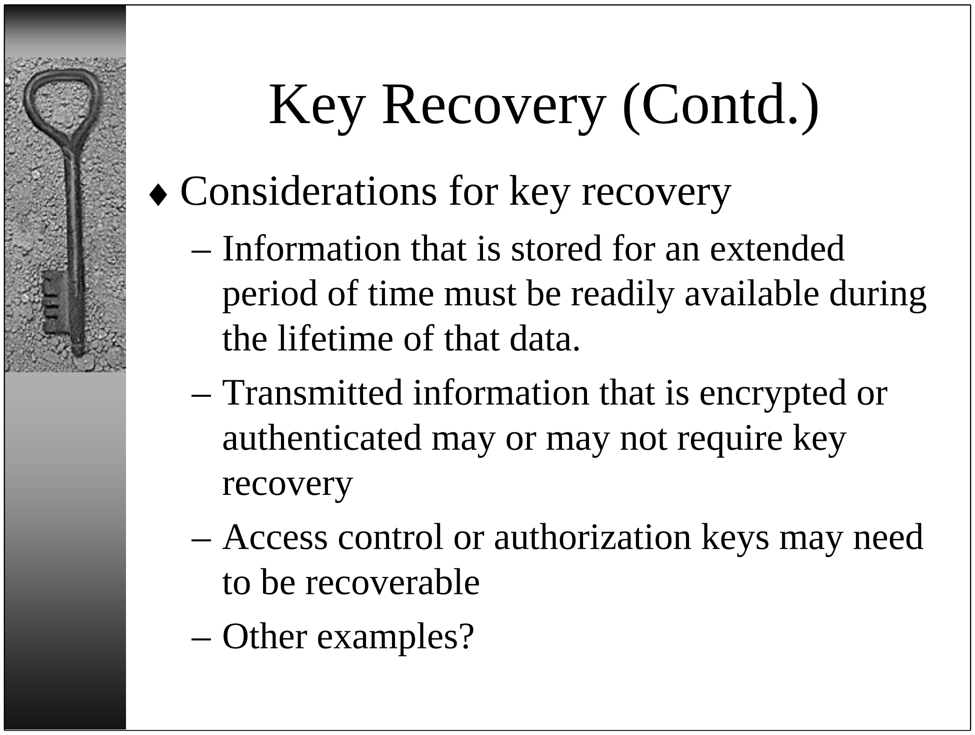
Key Recovery

- ♦ The process of retrieving keying material from backup or archive storage when it is not otherwise available
- ♦ Purpose: to recover (e.g., decrypt) or verify (e.g., authenticate) protected information on behalf of an organization or individual
- ♦ Use or non-use of key recovery should be a conscious decision

We’ve defined key recovery in the document to be…

We need key recovery in order…
If the data is encrypted or authenticated,
the decryption or authentication key must
also be available during the lifetime of the
data.

We feel that the...



Key Recovery (Contd.)

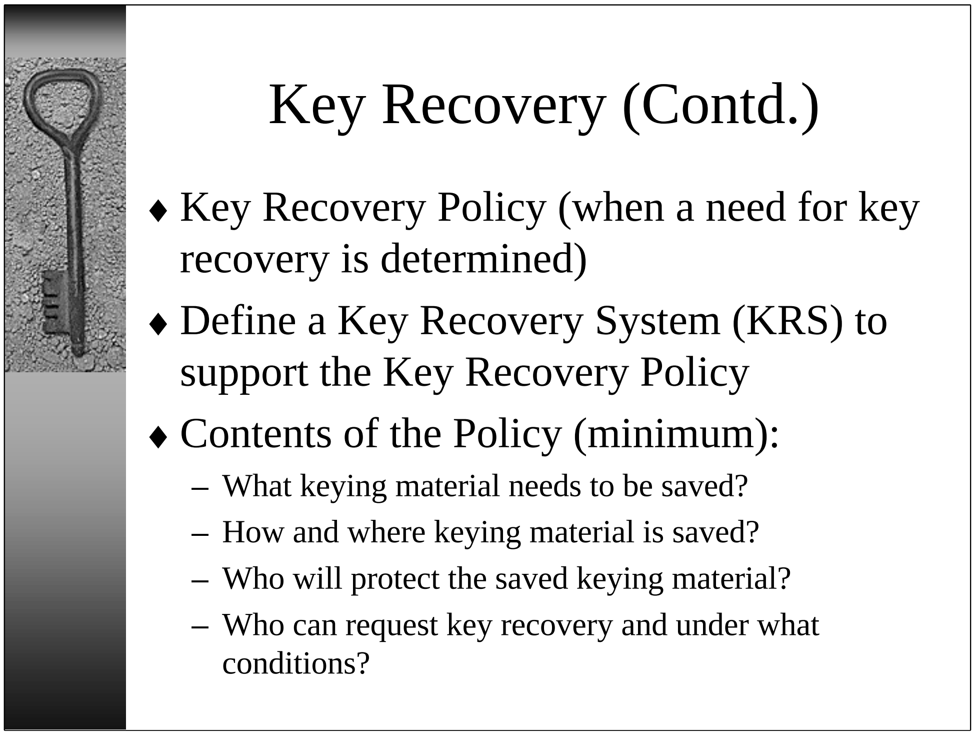
- ♦ Considerations for key recovery
  - – Information that is stored for an extended period of time must be readily available during the lifetime of that data.
  - – Transmitted information that is encrypted or authenticated may or may not require key recovery
  - – Access control or authorization keys may need to be recoverable
  - – Other examples?

We are considering providing some guidance or examples to consider when determining whether key recovery is needed.

1. Read bullet 1! If the data is encrypted or authenticated, the decryption or authentication key must also be available during the lifetime of the data.

2. Read bullet 2! Depending on the longevity of the information in its transmitted form.

3. Read bullet 3!

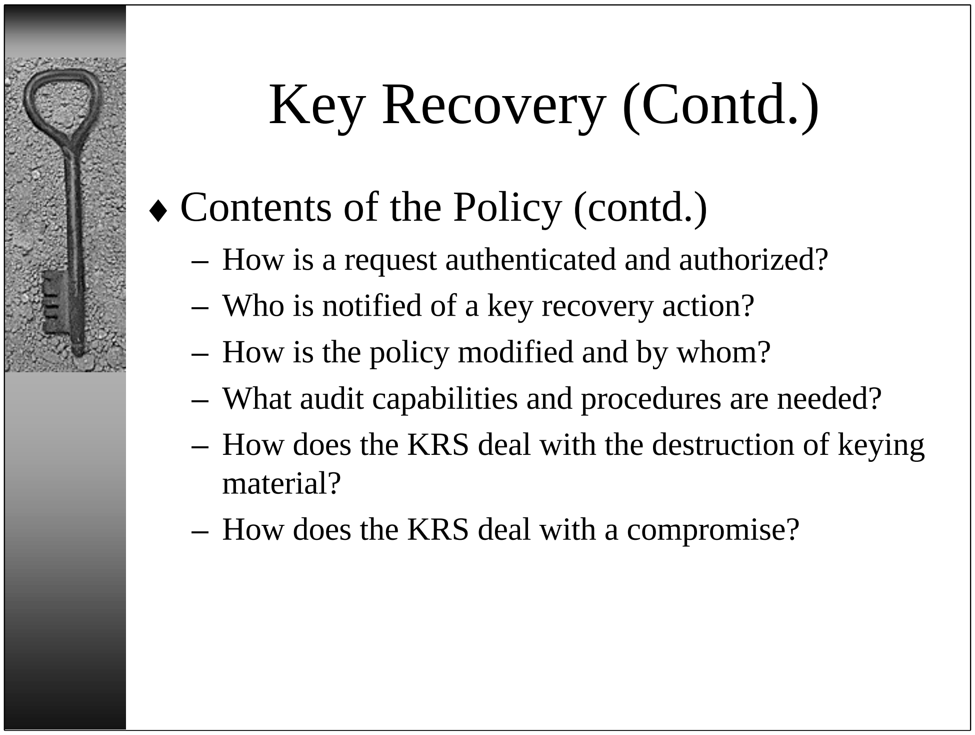


Key Recovery (Contd.)

- ♦ Key Recovery Policy (when a need for key recovery is determined)
- ♦ Define a Key Recovery System (KRS) to support the Key Recovery Policy
- ♦ Contents of the Policy (minimum):
  - – What keying material needs to be saved?
  - – How and where keying material is saved?
  - – Who will protect the saved keying material?
  - – Who can request key recovery and under what conditions?

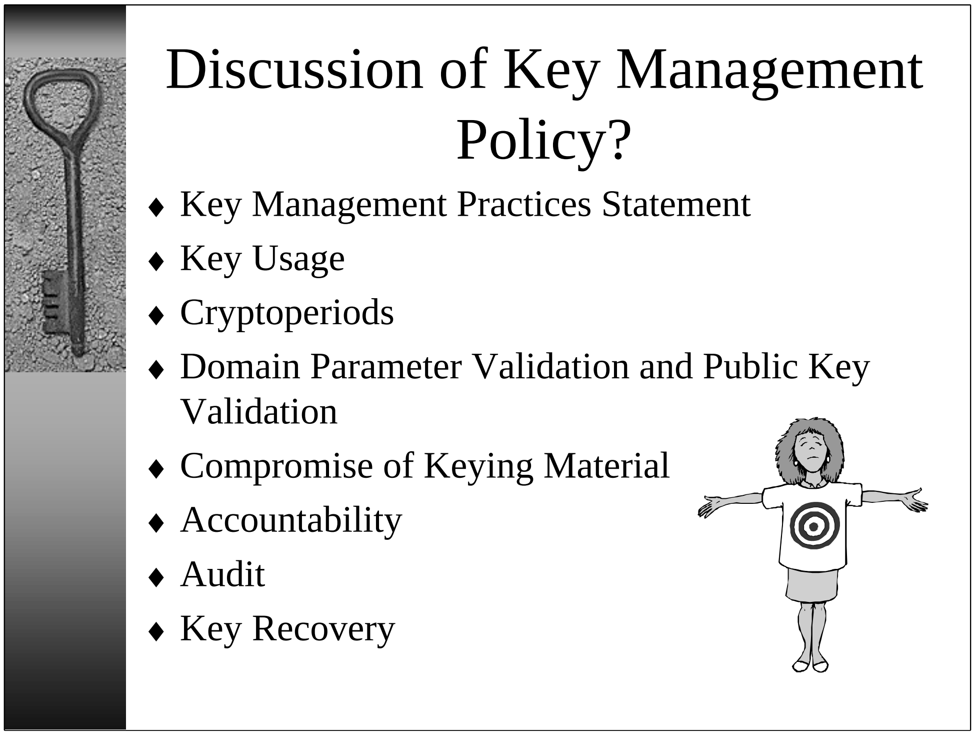
A Key Recovery Policy should be developed if key recovery is needed, and a Key Recovery System must be defined.

At a minimum the Key Recovery Policy should provide guidance on:...



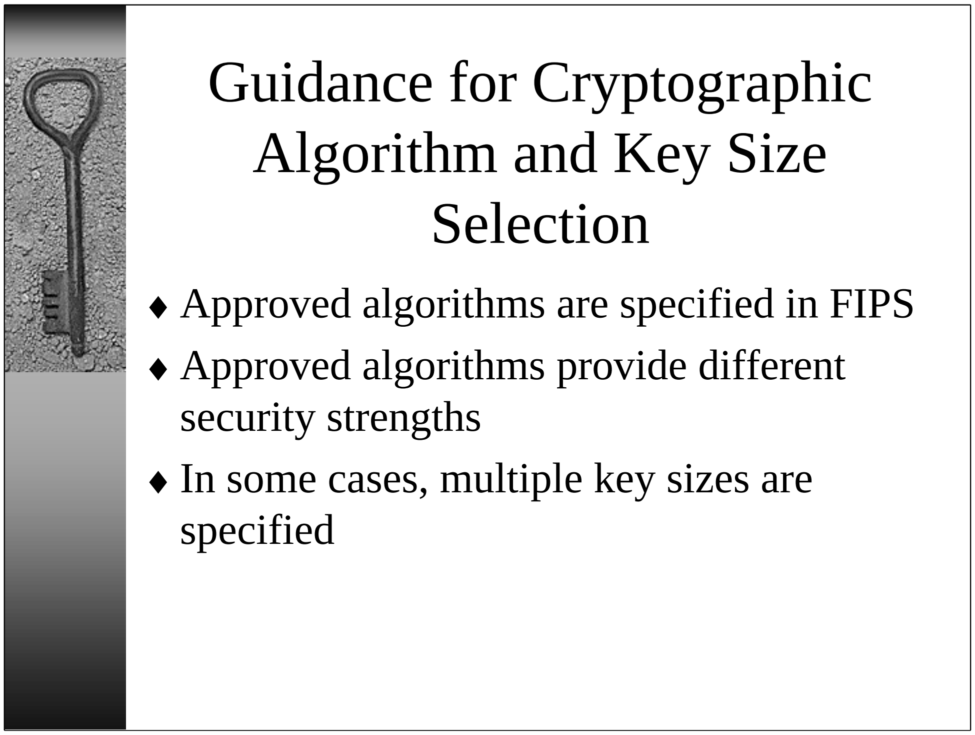
Key Recovery (Contd.)

- Contents of the Policy (contd.)
  - How is a request authenticated and authorized?
  - Who is notified of a key recovery action?
  - How is the policy modified and by whom?
  - What audit capabilities and procedures are needed?
  - How does the KRS deal with the destruction of keying material?
  - How does the KRS deal with a compromise?



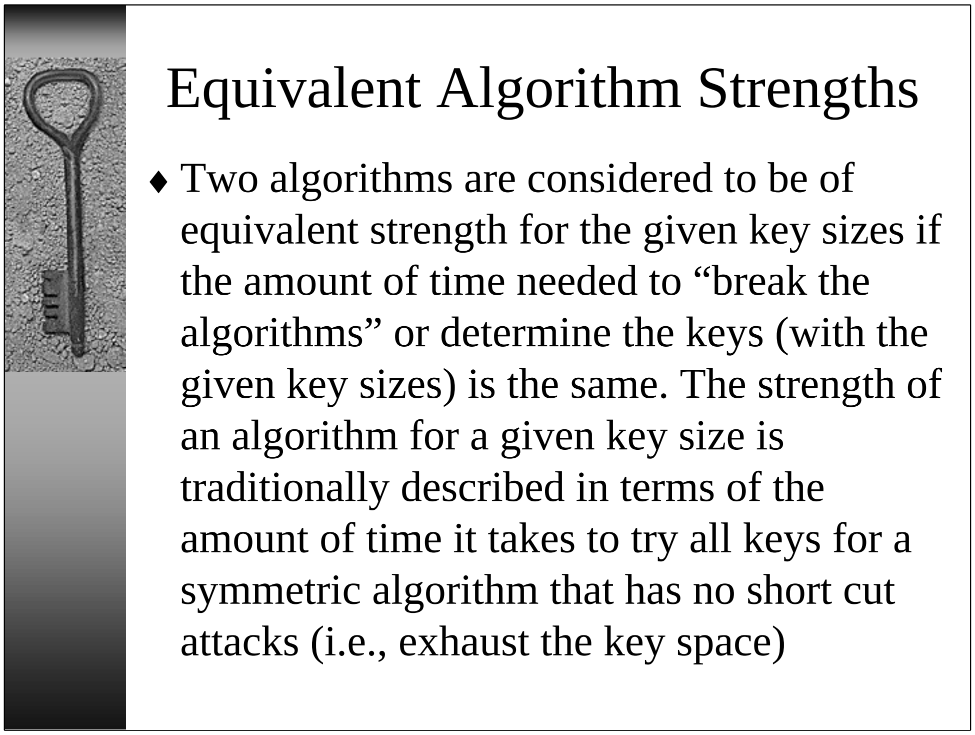
Discussion of Key Management Policy?

- ♦ Key Management Practices Statement
- ♦ Key Usage
- ♦ Cryptoperiods
- ♦ Domain Parameter Validation and Public Key Validation
- ♦ Compromise of Keying Material
- ♦ Accountability
- ♦ Audit
- ♦ Key Recovery



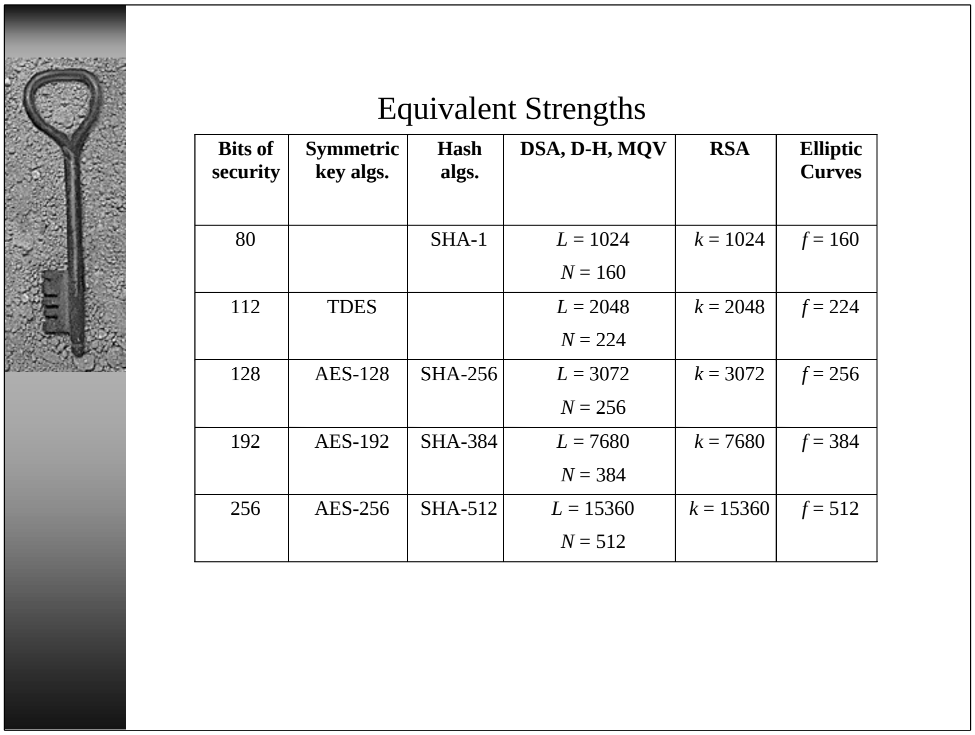
Guidance for Cryptographic Algorithm and Key Size Selection

- Approved algorithms are specified in FIPS
- Approved algorithms provide different security strengths
- In some cases, multiple key sizes are specified



Equivalent Algorithm Strengths

- ♦ Two algorithms are considered to be of equivalent strength for the given key sizes if the amount of time needed to “break the algorithms” or determine the keys (with the given key sizes) is the same. The strength of an algorithm for a given key size is traditionally described in terms of the amount of time it takes to try all keys for a symmetric algorithm that has no short cut attacks (i.e., exhaust the key space)



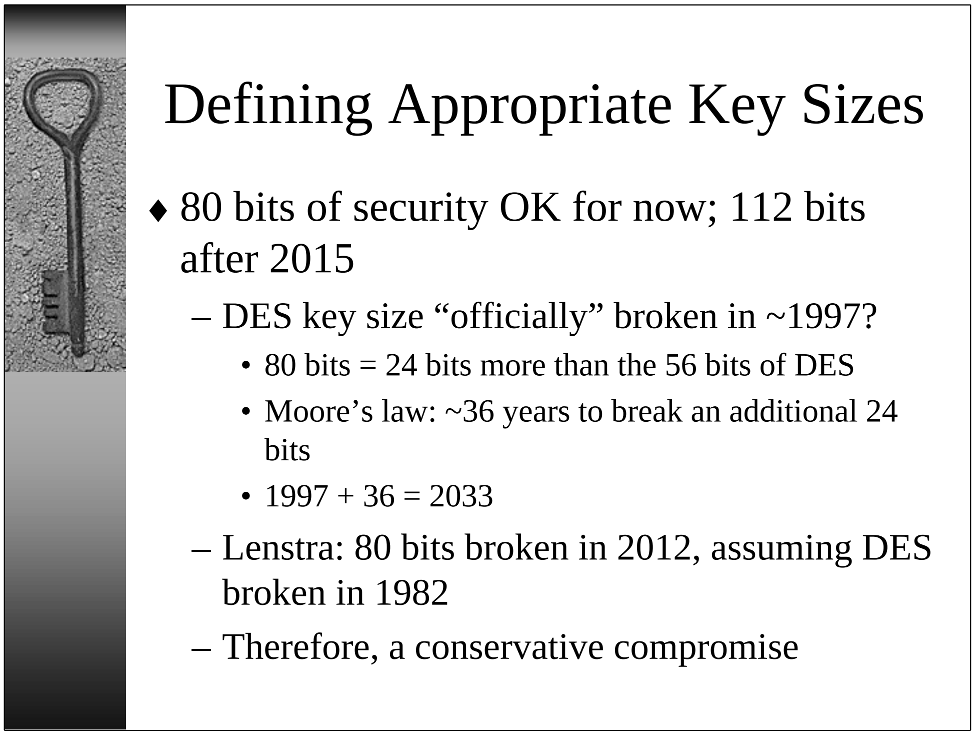
Equivalent Strengths

Bits of security	Symmetric key algs.	Hash algs.	DSA, D-H, MQV	RSA	Elliptic Curves
80		SHA-1	*L* = 1024 *N* = 160	*k* = 1024	*f* = 160
112	TDES		*L* = 2048 *N* = 224	*k* = 2048	*f* = 224
128	AES-128	SHA-256	*L* = 3072 *N* = 256	*k* = 3072	*f* = 256
192	AES-192	SHA-384	*L* = 7680 *N* = 384	*k* = 7680	*f* = 384
256	AES-256	SHA-512	*L* = 15360 *N* = 512	*k* = 15360	*f* = 512

Col. 4: *L* is the size of the modulus *p*, and *N* is the size of *q*. The value of *L* is considered to be the key size.

Col. 5: *k* is the size of the modulus *n*. The value of *k* is commonly considered to be the key size.

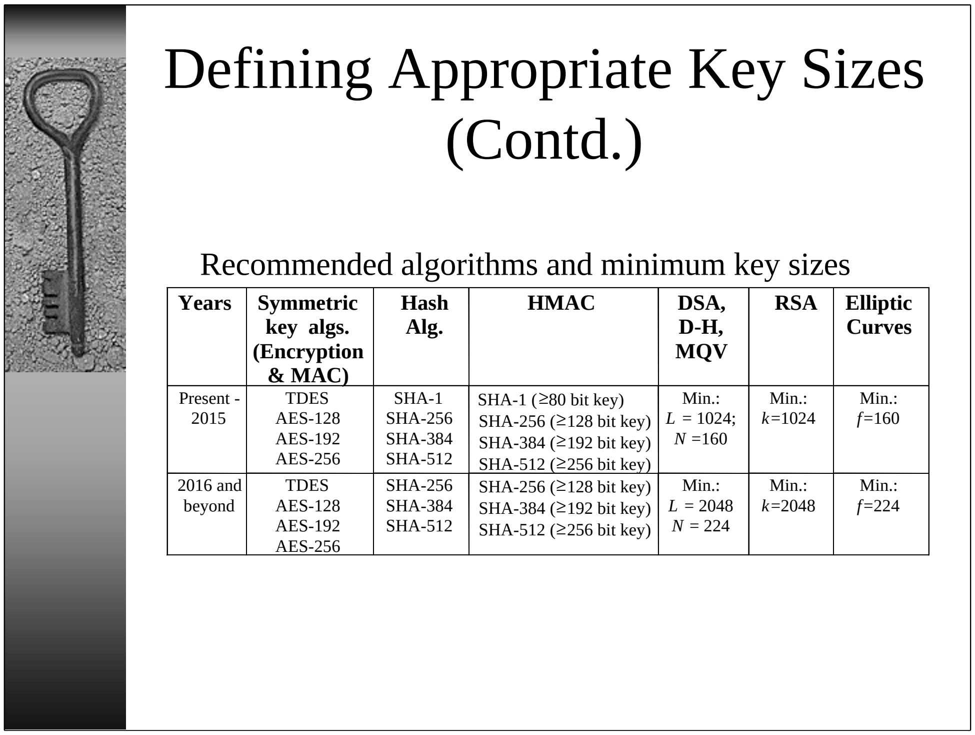
Col. 6: *f* is the size of *n*, where *n* is the order of the base point *G*. The value of *f* is commonly considered to be the key size.



Defining Appropriate Key Sizes

- ♦ 80 bits of security OK for now; 112 bits after 2015
  - – DES key size “officially” broken in ~1997?
    - 80 bits = 24 bits more than the 56 bits of DES
    - Moore’s law: ~36 years to break an additional 24 bits
    - 1997 + 36 = 2033
  - – Lenstra: 80 bits broken in 2012, assuming DES broken in 1982
  - – Therefore, a conservative compromise

Note: we could specify 128 bits of security
in 2015, but this would eliminate TDES.
Do we want to do this?
Here’s a way that we used to come up with 2015:
Moore's law states that the speed of processing
power will double every 18 months



Defining Appropriate Key Sizes (Contd.)

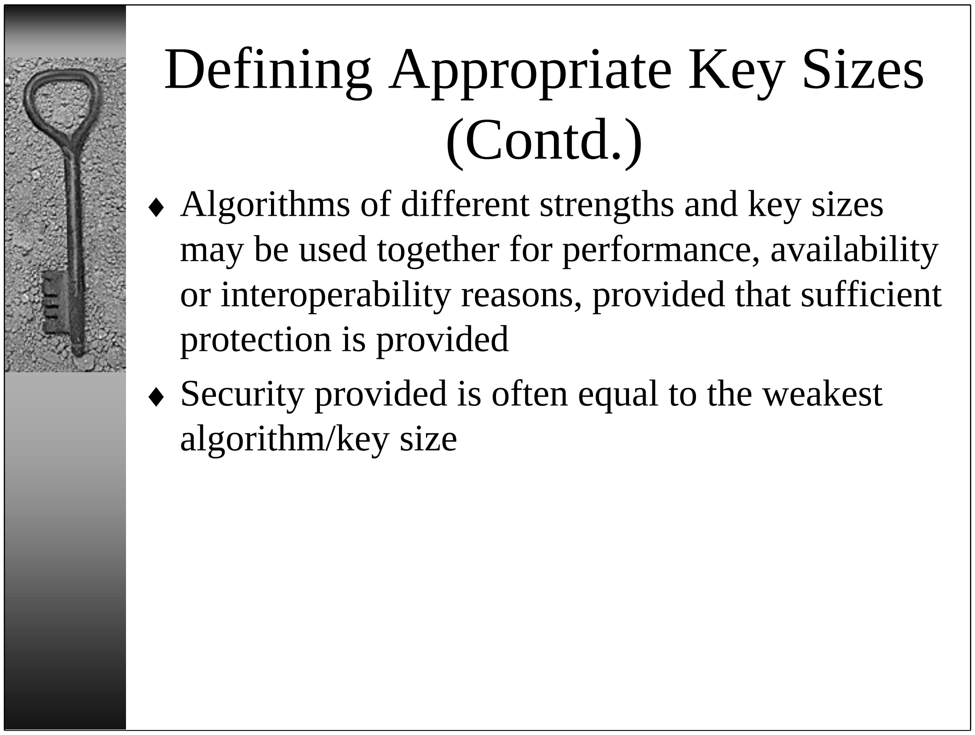
Recommended algorithms and minimum key sizes

Years	Symmetric key algs. (Encryption & MAC)	Hash Alg.	HMAC	DSA, D-H, MQV	RSA	Elliptic Curves
Present - 2015	TDES AES-128 AES-192 AES-256	SHA-1 SHA-256 SHA-384 SHA-512	SHA-1 (≥80 bit key) SHA-256 (≥128 bit key) SHA-384 (≥192 bit key) SHA-512 (≥256 bit key)	Min.: L = 1024; N =160	Min.: k=1024	Min.: f=160
2016 and beyond	TDES AES-128 AES-192 AES-256	SHA-256 SHA-384 SHA-512	SHA-256 (≥128 bit key) SHA-384 (≥192 bit key) SHA-512 (≥256 bit key)	Min.: L = 2048 N = 224	Min.: k=2048	Min.: f=224

Col. 5: L is the size of the modulus p, and N is the size of q. The value of L is considered to be the key size.

Col. 6: k is the size of the modulus n. The value of k is commonly considered to be the key size.

Col. 7: f is the size of n, where n is the order of the base point G. The value of f is commonly considered to be the key size.

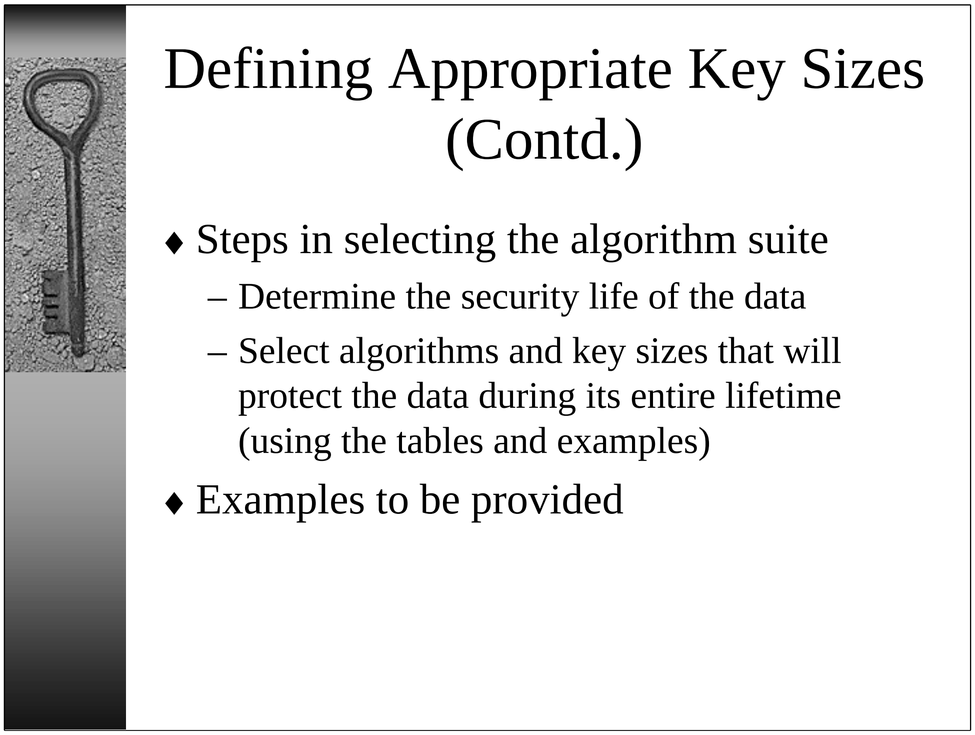


Defining Appropriate Key Sizes (Contd.)

- Algorithms of different strengths and key sizes may be used together for performance, availability or interoperability reasons, provided that sufficient protection is provided
- Security provided is often equal to the weakest algorithm/key size

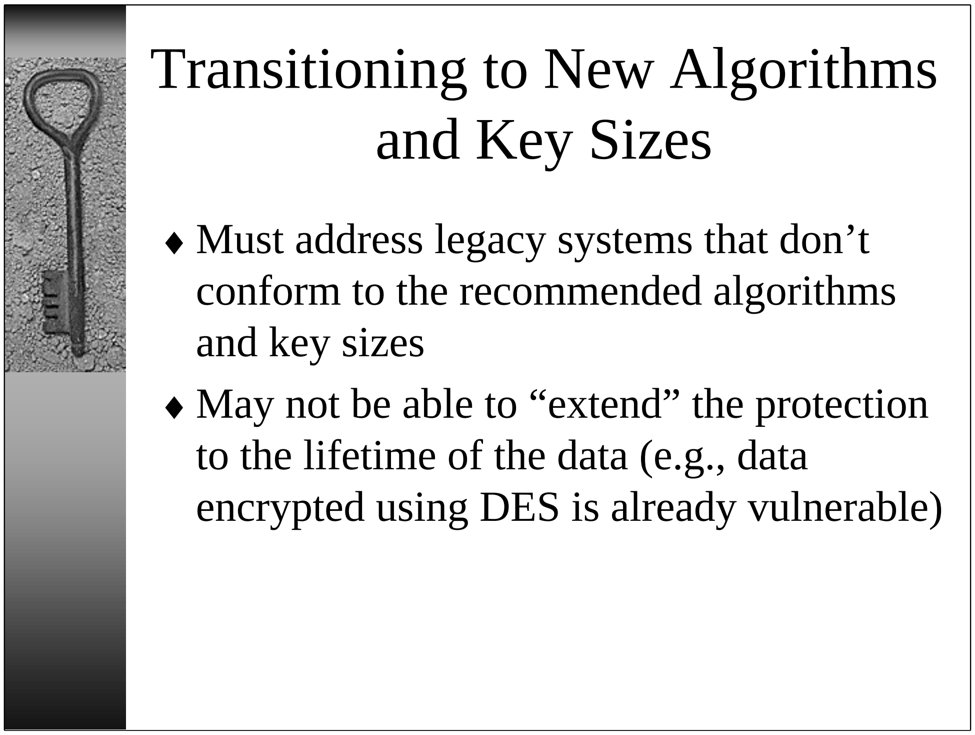
Example: when a key establishment algorithm having 80 bits of security is used to establish a 128 bit AES encryption key, only 80 bits of security are provided for the encrypted data)

Other examples to be provided



Defining Appropriate Key Sizes (Contd.)

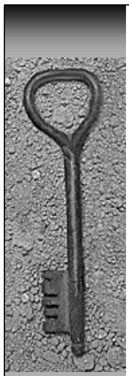
- ♦ Steps in selecting the algorithm suite
  - – Determine the security life of the data
  - – Select algorithms and key sizes that will protect the data during its entire lifetime (using the tables and examples)
- ♦ Examples to be provided



Transitioning to New Algorithms and Key Sizes

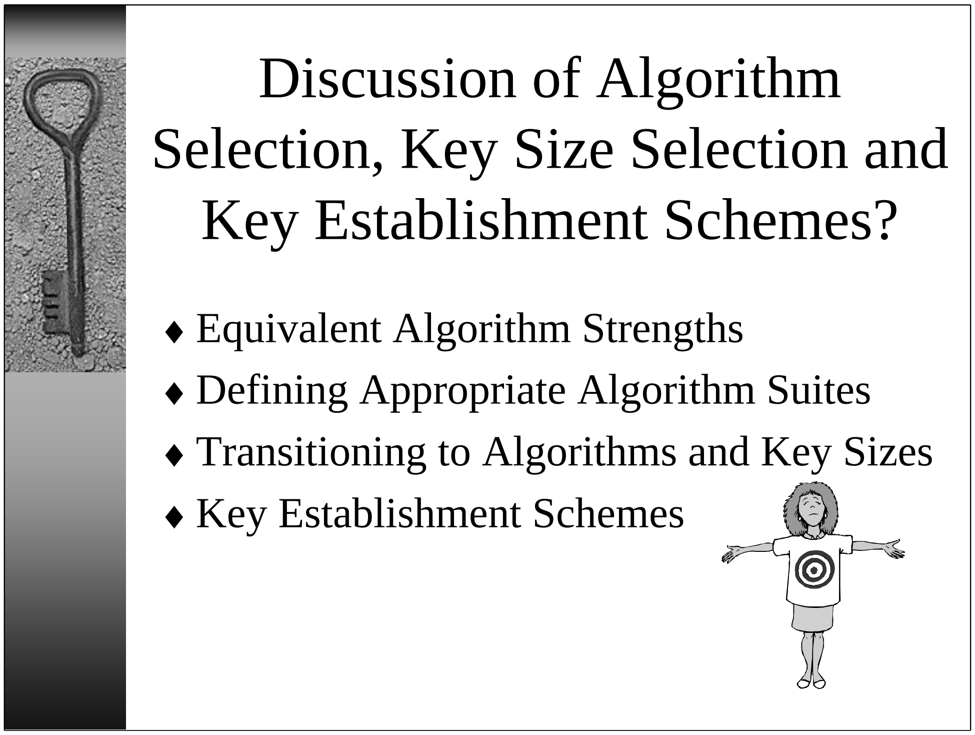
- Must address legacy systems that don’t conform to the recommended algorithms and key sizes
- May not be able to “extend” the protection to the lifetime of the data (e.g., data encrypted using DES is already vulnerable)

A warning may be the best that we can do.



Key Establishment Schemes

- Include additional guidance not included in the schemes document



Discussion of Algorithm Selection, Key Size Selection and Key Establishment Schemes?

- ♦ Equivalent Algorithm Strengths
- ♦ Defining Appropriate Algorithm Suites
- ♦ Transitioning to Algorithms and Key Sizes
- ♦ Key Establishment Schemes